Text as Data

Justin Grimmer

Associate Professor
Department of Political Science
Stanford University

October 2nd, 2014

Classification via Dictionary Methods

1) Task

Classification via Dictionary Methods

1) Task
    a) Categorize documents into predetermined categories

Classification via Dictionary Methods

1) Task
    a) Categorize documents into predetermined categories
    b) Measure documents association with predetermined categories

Classification via Dictionary Methods

1) Task
    a) Categorize documents into predetermined categories
    b) Measure documents association with predetermined categories
2) Objective function:

Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

where:

Classification via Dictionary Methods

1) Task
    a) Categorize documents into predetermined categories
    b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

where:

- $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_N)$ are word weights

Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

where:

- $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_N)$ are word weights
- $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iN})$ count the occurrence of each corresponding word in document i

Classification via Dictionary Methods

1) Task
   a) Categorize documents into predetermined categories
   b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

   where:
   - $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_N)$ are word weights
   - $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iN})$ count the occurrence of each corresponding word in document i
3) Optimization $\rightsquigarrow$ predetermined word list, no task specific optimization

Classification via Dictionary Methods

1) Task
    a) Categorize documents into predetermined categories
    b) Measure documents association with predetermined categories
2) Objective function:

$$f(\boldsymbol{\theta}, \boldsymbol{X}_i) = \frac{\sum_{j=1}^{N} \theta_j X_{ij}}{\sum_{j=1}^{N} X_{ij}}$$

where:

- $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_N)$ are word weights
- $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iN})$ count the occurrence of each corresponding word in document i

3) Optimization$\rightsquigarrow$ predetermined word list, no task specific optimization
4) Validation (Model checking)$\rightsquigarrow$ weight (model) checking, replication of hand coding, face validity

Word Weights: Separating Classes

General Classification Goal: Place documents into categories

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
    - Rely on Humans$\rightsquigarrow$ humans to identify words that associate with classes

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
    - Rely on Humans $\rightsquigarrow$ humans to identify words that associate with classes
    - Measure how well words separate (positive/negative, emotional, ...)

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
  - Rely on Humans $\rightsquigarrow$ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)
- Supervised Classification Methods (Later in the Quarter):

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
  - Rely on Humans$\rightsquigarrow$ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)
- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
  - Rely on Humans$\rightsquigarrow$ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)
- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
  - Rely on Humans$\leadsto$ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)
- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation

Word Weights: Separating Classes

General Classification Goal: Place documents into categories
How To Do Classification?

- Dictionaries:
  - Rely on Humans$\leadsto$ humans to identify words that associate with classes
  - Measure how well words separate (positive/negative, emotional, ...)
- Supervised Classification Methods (Later in the Quarter):
  - Rely on statistical models
  - Given set of coded documents, statistical relationship between classes/words
  - Statistical measures of separation

Key point: this is the same task

Types of Classification Problems

Topic: What is this text about?

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation

  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas

  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Sentiment: What is said in this text? [Public Opinion]

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Sentiment: What is said in this text? [Public Opinion]

- Positions on legislation
  $\Rightarrow$ { Support, Ambiguous, Oppose }
- Positions on Court Cases
  $\Rightarrow$ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
  $\Rightarrow$ { Liberal, Middle, Conservative, No Ideology Expressed }

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation

  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas

  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Sentiment: What is said in this text? [Public Opinion]

- Positions on legislation

  $\Rightarrow$ { Support, Ambiguous, Oppose }
- Positions on Court Cases

  $\Rightarrow$ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts

  $\Rightarrow$ { Liberal, Middle, Conservative, No Ideology Expressed }

Style/Tone: How is it said?

Types of Classification Problems

Topic: What is this text about?

- Policy area of legislation
  $\Rightarrow$ {Agriculture, Crime, Environment, ...}
- Campaign agendas
  $\Rightarrow$ {Abortion, Campaign, Finance, Taxing, ... }

Sentiment: What is said in this text? [Public Opinion]

- Positions on legislation
  $\Rightarrow$ { Support, Ambiguous, Oppose }
- Positions on Court Cases
  $\Rightarrow$ { Agree with Court, Disagree with Court }
- Liberal/Conservative Blog Posts
  $\Rightarrow$ { Liberal, Middle, Conservative, No Ideology Expressed }

Style/Tone: How is it said?

- Taunting in floor statements
  $\Rightarrow$ { Partisan Taunt, Intra party taunt, Agency taunt, ... }
- Negative campaigning
  $\Rightarrow$ { Negative ad, Positive ad}

Pre-existing word weights $\rightsquigarrow$ Dictionaries

Pre-existing word weights⇝ Dictionaries

`DICTION`

DICTION is a computer-aided text analysis program for Windows® and Mac® that uses a series of dictionaries to search a passage for five semantic features—Activity, Optimism, Certainty, Realism and Commonality—as well as thirty-five sub-features. DICTION uses predefined dictionaries and can use up to thirty custom dictionaries built with words that the user has defined, such as topical or negative words, for particular research needs.

Pre-existing word weights$\rightsquigarrow$ Dictionaries

DICTION

DICTION 7, now with *Power Mode,* can read a variety of text formats and can accept a large number of files within a single project. Projects containing over 1000 files are analyzed using *power analysis* for enhanced speed and reporting efficiency, with results automatically exported to .csv-formatted spreadsheet file.

Pre-existing word weights $\rightsquigarrow$ Dictionaries

DICTION

On an average computer, DICTION can process over 20,000 passages in about five minutes. DICTION requires 4.9 MB of memory and 38.4 MB of hard disk space.

Pre-existing word weights $\rightsquigarrow$ Dictionaries

DICTION

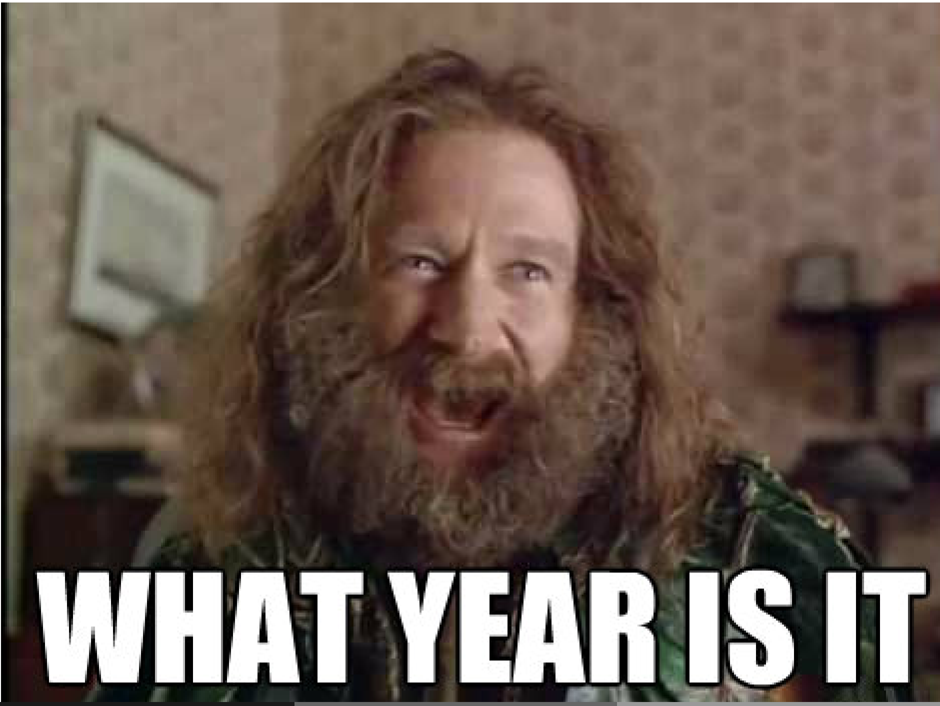
> "*provides both social scientific and humanistic understandings*"
>
> —Don Waisanen, Baruch College

Pre-existing word weights⇝ Dictionaries

`DICTION`

DICTION 7 for Mac (Educational) ($219.00)

This is the educational edition of DICTION Version 7 for Mac. You
purchase on the following page.


WHAT YEAR IS IT

Dictionary Methods

Many Dictionary Methods (like DICTION)

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary $\rightsquigarrow$ wrapped in GUI

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary $\rightsquigarrow$ wrapped in GUI
2) Basic tasks:

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary $\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
    a) Count words

Dictionary Methods

Many Dictionary Methods (like DICTION)

1) Proprietary $\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
    a) Count words
    b) Weighted counts of words

Dictionary Methods

Many Dictionary Methods (like DICTION)

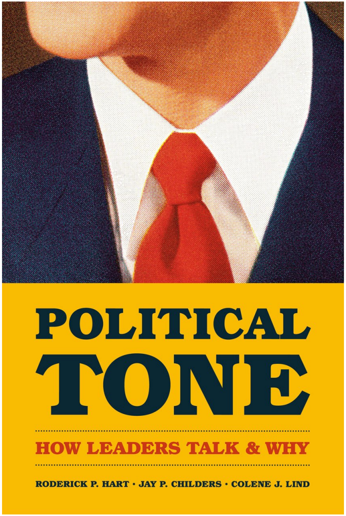
1) Proprietary $\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
    a) Count words
    b) Weighted counts of words
    c) Some graphics

Dictionary Methods

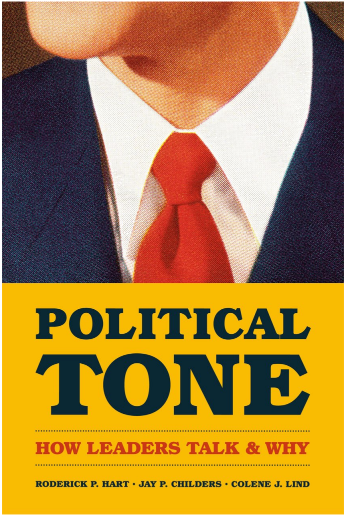
Many Dictionary Methods (like DICTION)

1) Proprietary$\rightsquigarrow$ wrapped in GUI
2) Basic tasks:
   a) Count words
   b) Weighted counts of words
   c) Some graphics
3) Pricey$\rightsquigarrow$ inexplicably

DICTION

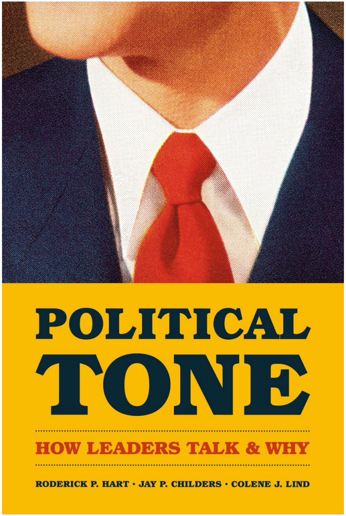

POLITICAL
TONE
HOW LEADERS TALK & WHY
RODERICK P. HART · JAY P. CHILDERS · COLENE J. LIND

DICTION



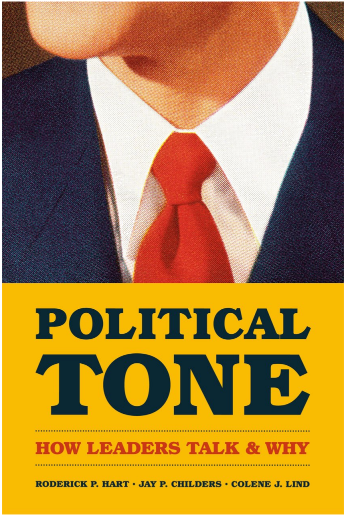
- { Certain, Uncertain }

DICTION



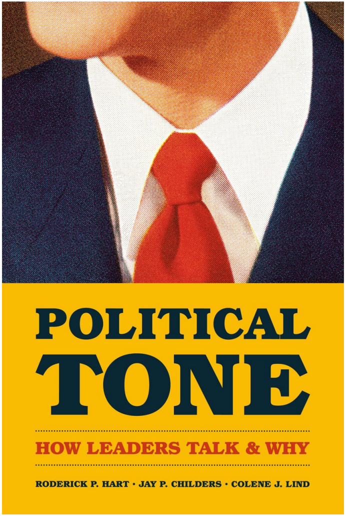
- { Certain, Uncertain }
  , { Optimistic, Pessimistic }

DICTION



- { Certain, Uncertain }, { Optimistic, Pessimistic }
- $\approx$ 10,000 words

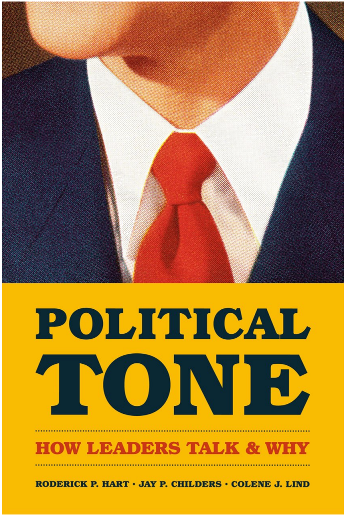
DICTION



- { Certain, Uncertain }, { Optimistic, Pessimistic }
- $\approx$ 10,000 words

Applies DICTION to a wide array of political texts

DICTION



- { Certain, Uncertain }
  , { Optimistic, Pessimistic }
- $\approx$ 10,000 words

Applies DICTION to a wide array of political texts
Examine specific periods of American political history

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)

Other Dictionaries

1) General Inquirer Database
   (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
    - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
    - { Positive, Negative }

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
    - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
    - { Positive, Negative }
    - 3627 negative and positive word strings
    - Workhorse for classification across many domains/papers

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
    - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
    - { Positive, Negative }
    - 3627 negative and positive word strings
    - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
    - Creation process:

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories$\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories$\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
     2) Judge round$\rightsquigarrow$ (a) Does the word belong? (b) What other categories might it belong to?

Other Dictionaries

1) General Inquirer Database
   (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories⇝ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
     2) Judge round⇝ (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories $\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
     2) Judge round $\rightsquigarrow$ (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories$\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
     2) Judge round$\rightsquigarrow$ (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes

- Harvard-IV-4

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
   - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
   - { Positive, Negative }
   - 3627 negative and positive word strings
   - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
   - Creation process:
     1) Generate word list for categories $\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
     2) Judge round $\rightsquigarrow$ (a) Does the word belong? (b) What other categories might it belong to?
   - { Positive emotion, Negative emotion }
   - 2300 words grouped into 70 classes

- Harvard-IV-4
- Affective Norms for English Words (we'll discuss this more later)

Other Dictionaries

1) General Inquirer Database (`http://www.wjh.harvard.edu/~inquirer/`)
    - Stone, P.J., Dumphy, D.C., and Ogilvie, D.M. (1966) *The General Inquirer: A Computer Approach to Content Analysis*
    - { Positive, Negative }
    - 3627 negative and positive word strings
    - Workhorse for classification across many domains/papers
2) Linguistic Inquiry Word Count (LIWC)
    - Creation process:
        1) Generate word list for categories$\rightsquigarrow$ " We drew on common emotion rating scales...Roget's Thesaurus...standard English dictionaries. [then] brain-storming sessions among 3-6 judges were held" to generate other words
        2) Judge round$\rightsquigarrow$ (a) Does the word belong? (b) What other categories might it belong to?
    - { Positive emotion, Negative emotion }
    - 2300 words grouped into 70 classes

- Harvard-IV-4
- Affective Norms for English Words (we'll discuss this more later)
- ...

Generating New Words

Three ways to create dictionaries (non-exhaustive):

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods$\rightsquigarrow$ next Tuesday

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods$\rightsquigarrow$ next Tuesday
- Manual generation

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output
  b) Mechanical turkers

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods$\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output
  b) Mechanical turkers
     - Example: $\{$ Happy, Unhappy $\}$

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods$\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output
  b) Mechanical turkers
     - Example: { Happy, Unhappy }
     - Ask turkers: how happy is

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output
  b) Mechanical turkers
     - Example: { Happy, Unhappy }
     - Ask turkers: how happy is `elevator`, `car`, `pretty`, `young`

Generating New Words

Three ways to create dictionaries (non-exhaustive):

- Statistical methods $\rightsquigarrow$ next Tuesday
- Manual generation
  - Careful thought (prayer? epiphanies? divine intervention?) about useful words
- Populations of people who are surprisingly willing to perform ill-defined tasks
  a) Undergraduates: Pizza $\rightarrow$ Research Output
  b) Mechanical turkers
     - Example: { Happy, Unhappy }
     - Ask turkers: how happy is
       `elevator`, `car`, `pretty`, `young`
       Output as dictionary

Applying Methods to Documents

Applying the model:

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}$, $(i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
    - $\theta_k \in \{0, 1\}$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \Re$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \, (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
    - $\theta_k \in \{0, 1\}$
    - $\theta_k \in \{-1, 0, 1\}$
    - $\theta_k \in \{-2, -1, 0, 1, 2\}$
    - $\theta_k \in \Re$

For each document i calculate score for document

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \Re$

For each document i calculate score for document

$$Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}$$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \Re$

For each document i calculate score for document

$$
\begin{aligned}
Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\
Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}}
\end{aligned}
$$

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
    - $\theta_k \in \{0, 1\}$
    - $\theta_k \in \{-1, 0, 1\}$
    - $\theta_k \in \{-2, -1, 0, 1, 2\}$
    - $\theta_k \in \Re$

For each document i calculate score for document

$$Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}$$

$$Y_i = \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \Re$

For each document i calculate score for document

$$Y_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k}$$

$$Y_i = \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}}$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
  - $\theta_k \in \{0, 1\}$
  - $\theta_k \in \{-1, 0, 1\}$
  - $\theta_k \in \{-2, -1, 0, 1, 2\}$
  - $\theta_k \in \Re$

For each document i calculate score for document

$$
\begin{aligned}
Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\
Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}}
\end{aligned}
$$

$Y_i \approx$ continuous $\rightsquigarrow$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

$Y_i < 0 \Rightarrow$ Negative Category

Applying Methods to Documents

Applying the model:

- Vector of word counts: $\boldsymbol{X}_i = (X_{i1}, X_{i2}, \ldots, X_{iK}, \ (i = 1, \ldots, N)$
- Weights attached to words $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$
    - $\theta_k \in \{0, 1\}$
    - $\theta_k \in \{-1, 0, 1\}$
    - $\theta_k \in \{-2, -1, 0, 1, 2\}$
    - $\theta_k \in \Re$

For each document i calculate score for document

$$
\begin{aligned}
Y_i &= \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_k} \\
Y_i &= \frac{\boldsymbol{\theta}^{'} \boldsymbol{X}_i}{\boldsymbol{X}_i^{'} \mathbf{1}}
\end{aligned}
$$

$Y_i \approx$ continuous $\leadsto$ Classification

$Y_i > 0 \Rightarrow$ Positive Category

$Y_i < 0 \Rightarrow$ Negative Category

$Y_i \approx 0$ Ambiguous

Applying a Dictionary to Press Releases

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website $\rightsquigarrow$ Theresa Wilson, Janyce Wiebe, and Paul Hoffman's dictionary

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website $\rightsquigarrow$ Theresa Wilson, Janyce Wiebe, and Paul Hoffman's dictionary
- Create positive/negative score for press releases.

Applying a Dictionary to Press Releases

- Collection of 169,779 press releases (US House members 2005-2010)
- Dictionary from Neal Caren's website $\rightsquigarrow$ Theresa Wilson, Janyce Wiebe, and Paul Hoffman's dictionary
- Create positive/negative score for press releases.

`Python` code and press releases

Examining Positive and Negative Statements in Press Releases

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009

Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010

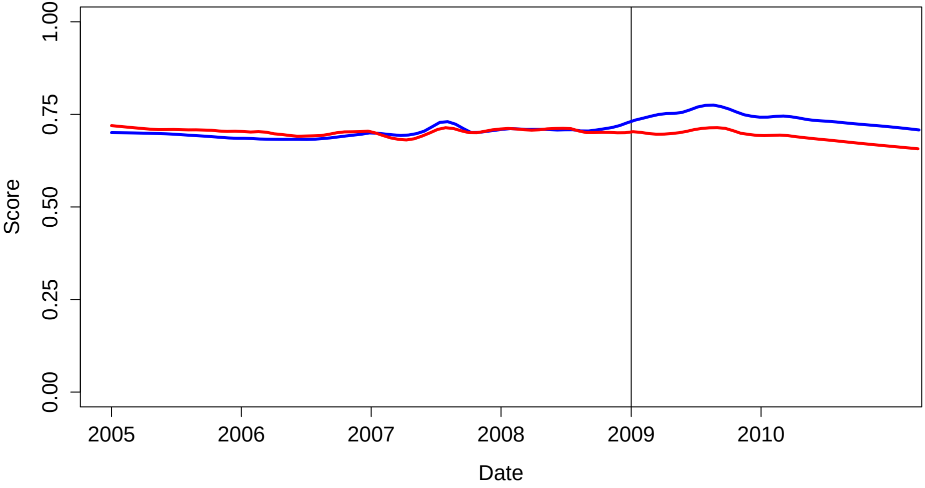
Examining Positive and Negative Statements in Press Releases

Least positive members of Congress:

1) Dan Burton, 2008
2) Nancy Pelosi, 2007
3) Mike Pence 2007
4) John Boehner, 2009
5) Jeff Flake, (basically all years)
6) Eric Cantor, 2009
7) Tom Price, 2010

Legislators who are more extreme $\rightsquigarrow$ less positive in press releases

Examining Positive and Negative Statements in Press Releases



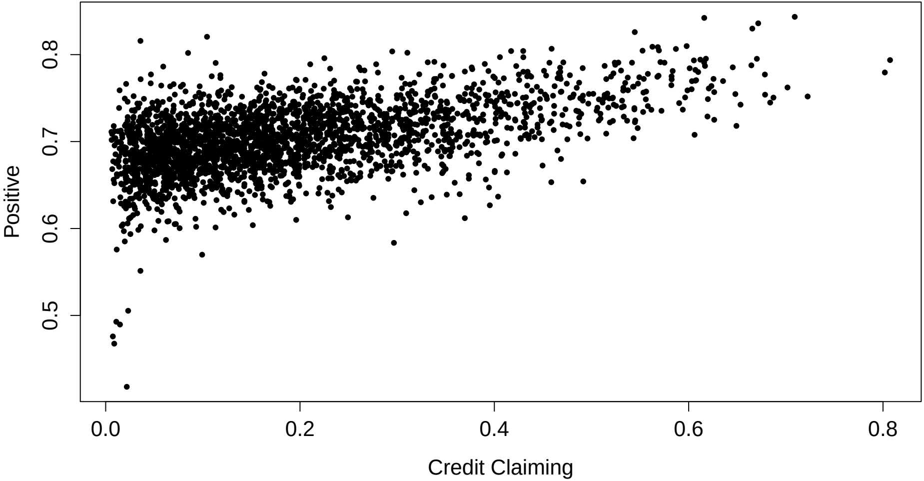
Examining Positive and Negative Statements in Press Releases

- Credit Claiming press release: 9.1 percentage points "more positive" than a non-credit claiming press release

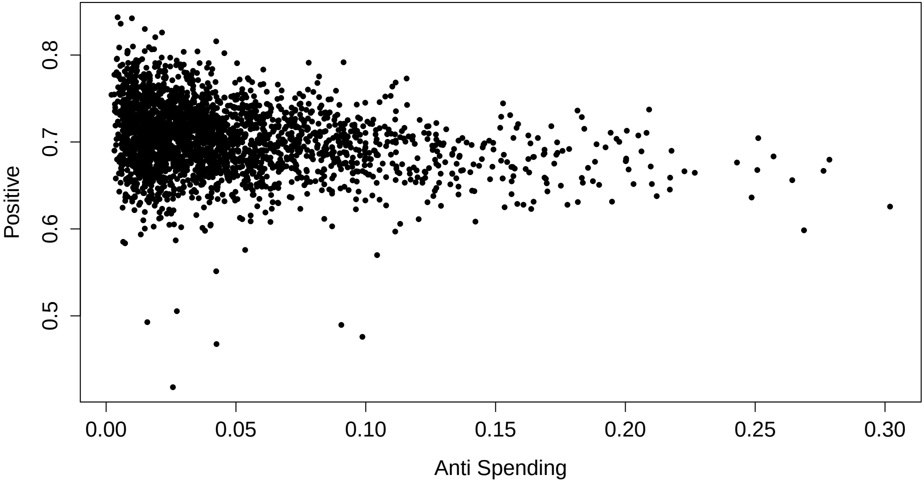
Examining Positive and Negative Statements in Press Releases

- Credit Claiming press release: 9.1 percentage points “more positive” than a non-credit claiming press release
- Anti-spending press release: 10.6 percentage points “less positive” than a non-anti spending press release

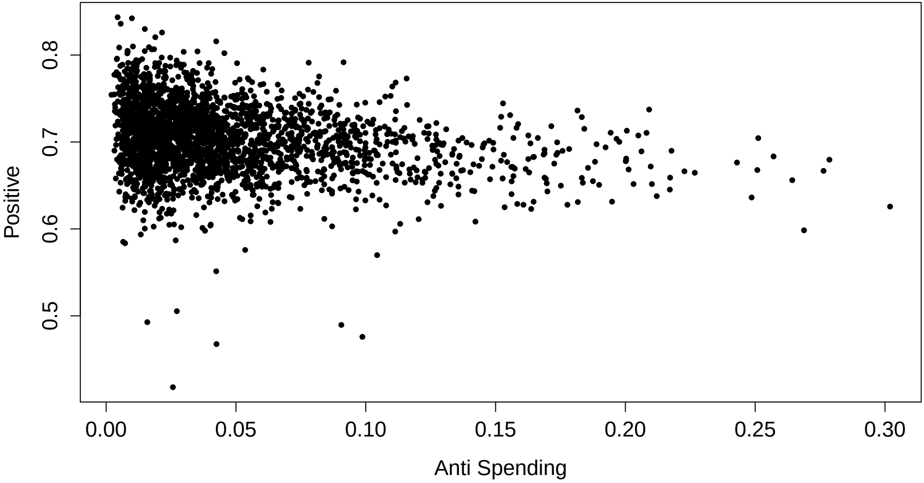
Examining Positive and Negative Statements in Press Releases



Examining Positive and Negative Statements in Press Releases



Examining Positive and Negative Statements in Press Releases



Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context
- Without optimization$\rightsquigarrow$ unclear about dictionaries performance

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context
- Without optimization$\rightsquigarrow$ unclear about dictionaries performance

Just because dictionaries provide measures labeled "positive" or "negative" it doesn't mean they are accurate measures in your text (!!!!)

Methodological Issues/Problems with Dictionaries

Dictionary methods are context invariant

- No optimization step $\rightsquigarrow$ same word weights regardless of texts
- Optimization$\rightsquigarrow$ incorporate information specific to context
- Without optimization$\rightsquigarrow$ unclear about dictionaries performance

Just because dictionaries provide measures labeled "positive" or "negative" it doesn't mean they are accurate measures in your text (!!!!)

Validation

Validation

Classification Validity:

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out?

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
    - Is the classification scheme well defined for your texts?
    - Can humans accomplish the coding task?
    - Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out? Over fitting

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out? Over fitting
- Using off-the-shelf dictionary: all labeled documents to test

Validation

Classification Validity:

- Training: build dictionary on subset of documents with known labels
- Test: apply dictionary method to other documents with known labels
- Requires hand coded documents
- Hand coded documents useful for other reasons
  - Is the classification scheme well defined for your texts?
  - Can humans accomplish the coding task?
  - Is the dictionary your using appropriate?

Replicate classification exercise

- How well does our method perform on held out documents?
- Why held out? Over fitting
- Using off-the-shelf dictionary: all labeled documents to test
- Supervised learning classification: (Cross)validation

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
    - Ambiguity in language
    - Limited working memory

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
    - Ambiguity in language
    - Limited working memory
    - Ambiguity in classification rules

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
  - Ambiguity in language
  - Limited working memory
  - Ambiguity in classification rules
- A procedure for training coders:

Hand Coding: A Brief Digression

Humans should be able to classify documents into the categories you want the machine to classify them in

- This is hard
- Why?
    - Ambiguity in language
    - Limited working memory
    - Ambiguity in classification rules
- A procedure for training coders:
    1) Coding rules
    2) Apply to new texts
    3) Assess coder agreement (we'll discuss more in a few weeks)
    4) Using information and discussion, revise coding rules

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

$$\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}$$

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

$$\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}$$

$$\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}$$

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

$$\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}$$

$$\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}$$

$$\text{Recall}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}}$$

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

$$\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}$$

$$\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}$$

$$\text{Recall}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}}$$

$$F_{\text{Liberal}} = \frac{2\text{Precision}_{\text{Liberal}}\text{Recall}_{\text{Liberal}}}{\text{Precision}_{\text{Liberal}} + \text{Recall}_{\text{Liberal}}}$$

Assessing Classification

Measures of classification performance

	Actual Label	
Guess	Liberal	Conservative
Liberal	True Liberal	False Liberal
Conservative	False Conservative	True Conservative

$$\text{Accuracy} = \frac{\text{TrueLib} + \text{TrueCons}}{\text{TrueLib} + \text{TrueCons} + \text{FalseLib} + \text{FalseCons}}$$

$$\text{Precision}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Liberal}}$$

$$\text{Recall}_{\text{Liberal}} = \frac{\text{True Liberal}}{\text{True Liberal} + \text{False Conservative}}$$

$$F_{\text{Liberal}} = \frac{2\text{Precision}_{\text{Liberal}}\text{Recall}_{\text{Liberal}}}{\text{Precision}_{\text{Liberal}} + \text{Recall}_{\text{Liberal}}}$$

Under reported for dictionary classification

What about continuous measures?

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point$\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification $\rightsquigarrow$ label phrases and then aggregate

$\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification $\rightsquigarrow$ label phrases and then aggregate
Modifiable areal unit problem in texts $\rightsquigarrow$

What about continuous measures?

Necessarily more complicated

- Go back to hand coding exercise
- Imagine asking undergraduates to rate document on a continuous scale (0-100)
- Difficult to create classifications with agreement
- Precisely the point $\rightsquigarrow$ merely creating a gold standard is hard, let alone computer classification

Lower level classification $\rightsquigarrow$ label phrases and then aggregate
Modifiable areal unit problem in texts $\rightsquigarrow$ aggregating destroys information, conclusion may depend on level of aggregation

Validation, Dictionaries from other Fields

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting:

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting: `tax,cost,capital,board,liability,foreign, cancer, crude(oil),tire`

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting: `tax,cost,capital,board,liability,foreign, cancer, crude(oil),tire`
- 73% of Harvard negative words in this set(!!!!!)

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting: `tax,cost,capital,board,liability,foreign, cancer, crude(oil),tire`
- 73% of Harvard negative words in this set(!!!!!)
- Not Negative Harvard, Negative in Accounting:

Validation, Dictionaries from other Fields

Accounting Research: measure tone of 10-K reports

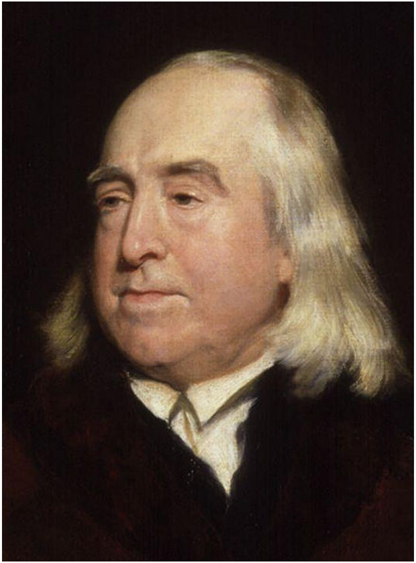
- tone matters ($)

Previous state of art: Harvard-IV-4 Dictionary applied to texts

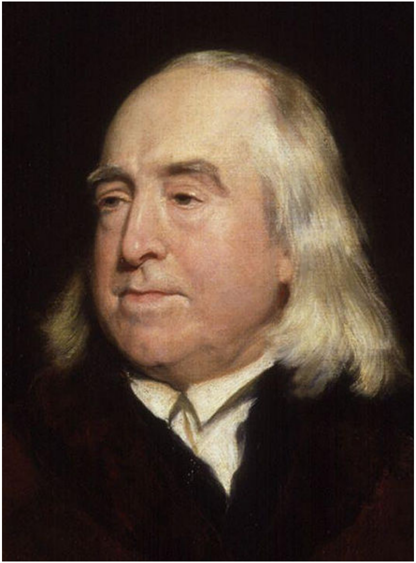
Loughran and McDonald (2011): Financial Documents are Different, polysemes

- Negative words in Harvard, Not Negative in Accounting: `tax,cost,capital,board,liability,foreign, cancer, crude(oil),tire`
- 73% of Harvard negative words in this set(!!!!!)
- Not Negative Harvard, Negative in Accounting: `felony,litigation,restated,misstatement, andunanticipated`

Measuring Happiness

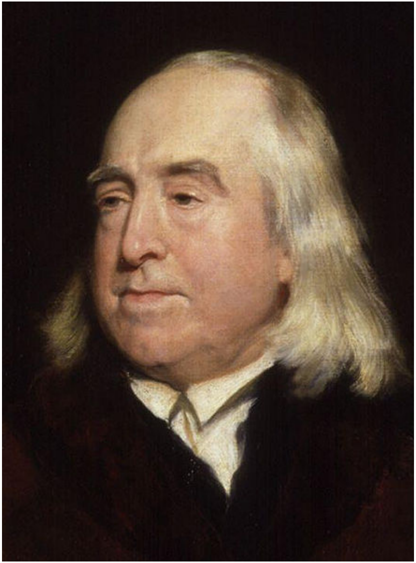


Measuring Happiness



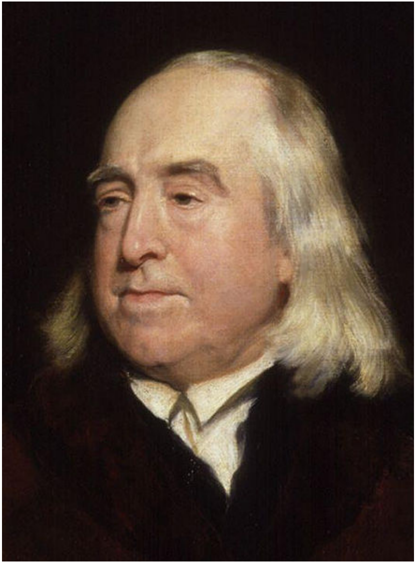
- Quantifying Happiness: How happy is society?

Measuring Happiness



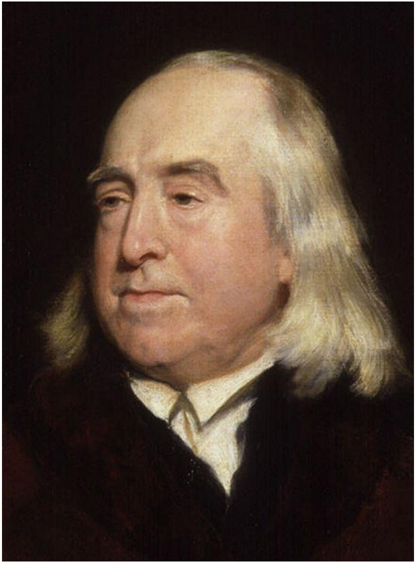
- Quantifying Happiness: How happy is society?
- How Happy is a Song?

Measuring Happiness



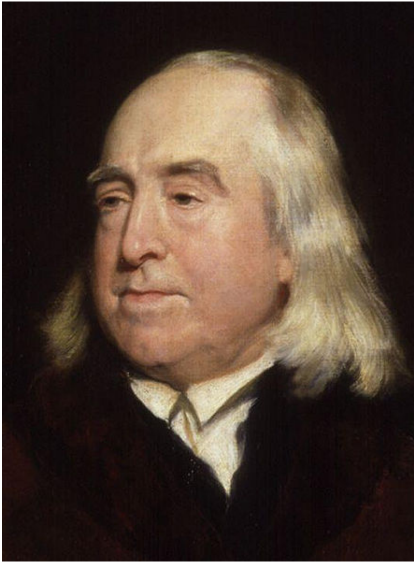
- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?

Measuring Happiness



- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)

Measuring Happiness



- Quantifying Happiness: How happy is society?
- How Happy is a Song?
- Blog posts?
- Facebook posts? (Gross National Happiness)

Use Dictionary Methods

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
    - On a scale of 1-9 how happy does this word make you?
      Happy : triumphant (8.82)/paradise (8.72)/ love (8.72)

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy : triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy : triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    Unhappy : cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy : triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    Unhappy : cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)
- Happiness for text i (with word j having happiness θ_j and document frequence X_{ij}

Measuring Happiness

Dodds and Danforth (2009): Use a dictionary method to measure happiness

- Affective Norms for English Words (ANEW)
- Bradley and Lang 1999: 1034 words, Affective reaction to words
  - On a scale of 1-9 how happy does this word make you?
    Happy : triumphant (8.82)/paradise (8.72)/ love (8.72)
    Neutral: street (5.22)/ paper (5.20)/ engine (5.20)
    Unhappy : cancer (1.5)/funeral (1.39)/ rape (1.25) /suicide (1.25)
- Happiness for text i (with word j having happiness θ_j and document frequence X_{ij}

$$\text{Happiness}_i = \frac{\sum_{k=1}^{K} \theta_k X_{ik}}{\sum_{k=1}^{K} X_{ik}}$$

Lyrics for
Michael Jackson's Billie Jean

"She was more like a beauty queen
from a movie scene.
⋮
And mother always told me,
be careful who you love.
And be careful of what you do
'cause the lie becomes the truth.
Billie Jean is not my lover,
She's just a girl who claims
that I am the one.
⋮

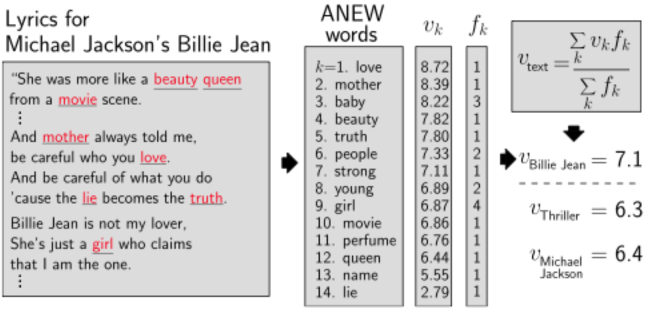
ANEW words	v_k	f_k
k=1. love	8.72	1
2. mother	8.39	1
3. baby	8.22	3
4. beauty	7.82	1
5. truth	7.80	1
6. people	7.33	2
7. strong	7.11	1
8. young	6.89	2
9. girl	6.87	4
10. movie	6.86	1
11. perfume	6.76	1
12. queen	6.44	1
13. name	5.55	1
14. lie	2.79	1

$$v_{\text{text}} = \frac{\sum_k v_k f_k}{\sum_k f_k}$$

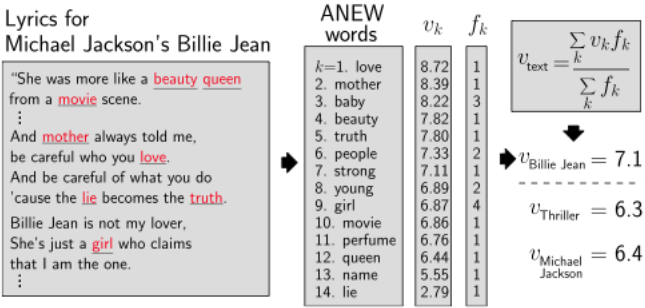
$$v_{\text{Billie Jean}} = 7.1$$

$$v_{\text{Thriller}} = 6.3$$

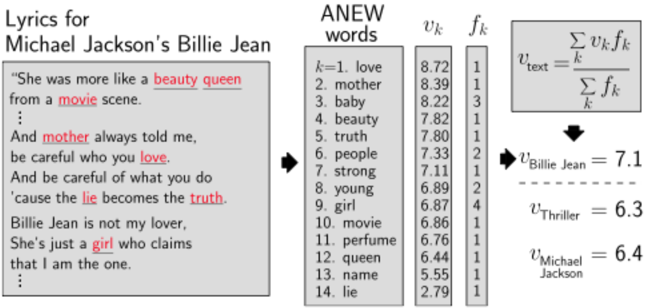
$$v_{\text{Michael Jackson}} = 6.4$$



Homework Hints: One approach: write a `for` loop searching for words in dictionary (caution: is dictionary stemmed?)

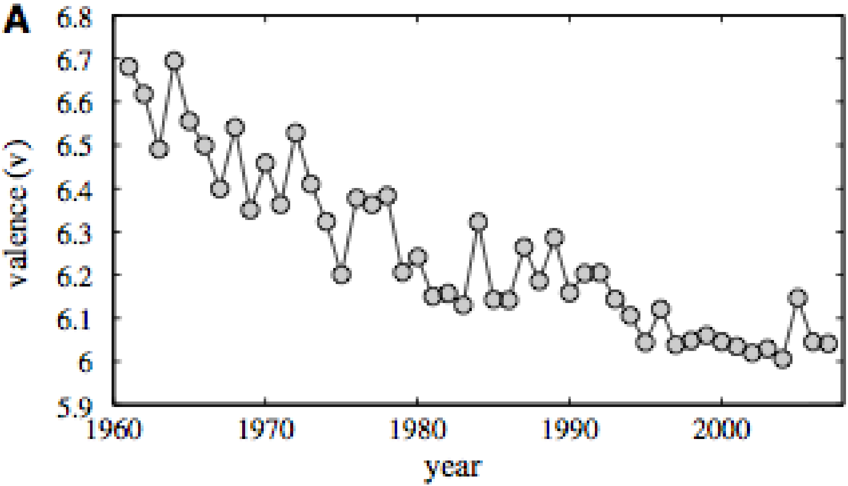


Homework Hints: One approach: write a `for` loop searching for words in dictionary (caution: is dictionary stemmed?)
Happiest Song on Thriller?

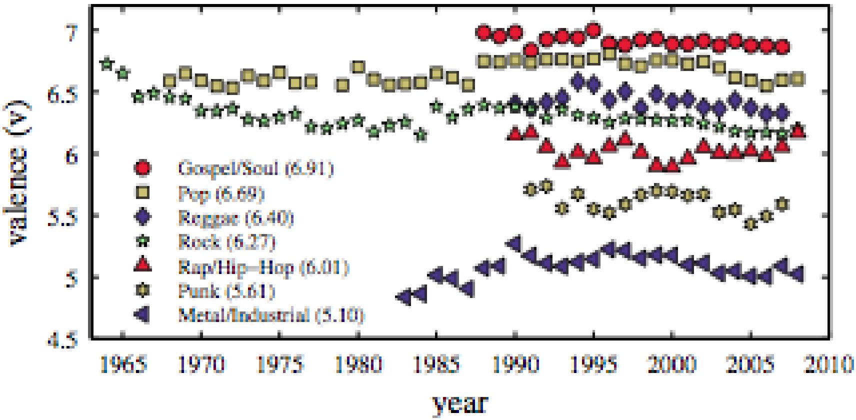


Homework Hints: One approach: write a `for` loop searching for words in dictionary (caution: is dictionary stemmed?)
Happiest Song on Thriller?
P.Y.T. (Pretty Young Thing) (This is the right answer!)

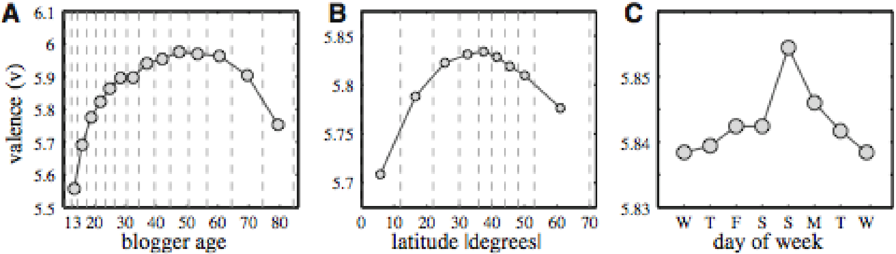
Happiness in Society



Happiness in Society



Happiness in Society



Dictionary Methods

Today: Classification via Dictionaries
Next week: Seperating Words and the Geometry of Text
Good luck on the homework!